

(19) KOREAN INTELLECTUAL PROPERTY OFFICE

KOREAN PATENT ABSTRACTS

(11)Publication number: **1020030027672 A**
 (43)Date of publication of application: **07.04.2003**

(21)Application number: **1020020057393**
 (22)Date of filing: **19.09.2002**
 (30)Priority: **27.09.2001 1**

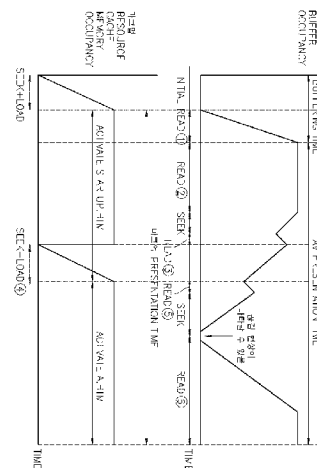
(71)Applicant: **SAMSUNG ELECTRONICS CO., LTD.**
 (72)Inventor: **HUH, JEONG GWON
 JUNG, HYEON GWON
 KO, JEONG WAN**

(51)Int. Cl **G11B 20/10**

(54) INFORMATION RECORDING MEDIA WITH RECORDED PRELOAD INFORMATION, AND DEVICE AND METHOD THEREOF

(57) Abstract:

PURPOSE: An information recording medium with recorded preload information, and a method and a device thereof are provided to prevent the disconnection of motion picture screen and recognize files to be preloaded or deleted for effective preloading or deleting. **CONSTITUTION:** An information recording medium with recorded preload information includes AV data including audio and/or video data, and a markup document inserted with preload information for commanding the display of the AV data, and reading and storing a file to be preloaded. The markup document prevents the disconnection of motion picture screen.



copyright KIPO 2003

Legal Status

Date of request for an examination (20030203)
 Notification date of refusal decision (00000000)
 Final disposal of an application (registration)
 Date of final disposal of an application (20060111)
 Patent registration number (1005538810000)
 Date of registration (20060214)
 Number of opposition against the grant of a patent ()
 Date of opposition against the grant of a patent (0 00000000)
 Number of trial against decision to refuse ()
 Date of requesting trial against decision to refuse ()
 Date of extinction of right ()

(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(51) 。 Int. Cl. ⁷
G11B 20/10

(11) 공개번호 특2003- 0027672
(43) 공개일자 2003년04월07일

(21) 출원번호 10- 2002- 0057393
(22) 출원일자 2002년09월19일

(30) 우선권주장 1020010060137 2001년09월27일 대한민국(KR)
1020010065393 2001년10월23일 대한민국(KR)

(71) 출원인 삼성전자주식회사
경기도 수원시 팔달구 매탄3동 416번지

(72) 발명자 정현권
경기도광주군광주읍탄벌리동보아파트104동906호
고정완
경기도수원시팔달구망포동벽산아파트103동201호
허정권
서울특별시서초구반포2동주공아파트2단지203동504호

(74) 대리인 이영필
이해영

심사청구 : 있음

(54) 프리로드 정보가 기록된 정보저장매체, 그 재생장치 및재생방법

요약

본 발명은 프리로드 정보가 기록된 정보저장매체, 그 재생장치 및 재생방법에 관한 것이다.

본 발명에 따른 정보저장매체는 오디오 및/또는 비디오 데이터를 포함하는 AV 데이터; 및 디코딩되어 재생된 상기 AV 데이터를 디스플레이하고, 프리로드 대상 파일을 독출하여 메모리에 저장할 것을 명령하는 프리로드 정보가 삽입되어 있는 마크업 문서를 포함하는 것을 특징으로 한다. 이에 의해, DVD에 기록된 AV 데이터를 재생하여 마크업 문서를 통해 디스플레이하는 경우 동영상 화면의 끊김현상이 발생되지 않도록 하는 마크업 문서가 기록된 정보저장매체, 그 재생장치 및 재생방법이 제공된다.

대표도

도 2

명세서

도면의 간단한 설명

도 1은 AV 데이터가 기록된 인터랙티브 DVD의 개략도,

도 2는 도 1의 인터랙티브 DVD를 재생하는 과정에서 발생할 수 있는 끊김현상을 설명하기 위한 참고도,

도 3은 본 발명의 바람직한 실시예에 따른 재생장치의 블록도,

도 4a는 DVD(300)의 파일 구조의 일 예를 도시한 참고도,

도 4b는 DVD(300)의 파일 구조의 다른 예를 도시한 참고도,

도 5a는 DVD(300)의 볼륨 공간(Volume Space)의 일 개략도,

도 5b는 DVD(300)의 볼륨 공간(Volume Space)의 다른 개략도,

도 6은 본 발명에 따른 프리로드 방식을 사용자 인터페이스 관점에서 설명하기 위한 참고도,

도 6은 본 발명의 바람직한 실시예에 따른 재생방법을 설명하기 위한 플로우차트,

도 7은 도 6의 프리로드 정보 해석단계(602단계)의 일 구현예,

도 8은 도 6의 프리로드 대상 파일의 프리로드 실행단계(603단계)의 일 구현예,

도 9a는 도 6의 프리로드 대상 파일의 프리로드 실행단계(603단계)의 다른 구현예,

도 9b는 도 6의 프리로드 대상 파일의 프리로드 실행단계(603단계)의 또 다른 구현예,

도 10은 프리로드 리스트 파일에 프리로드 대상 파일의 크기가 포함되어 있을 경우 프리로드하는 방법을 설명하기 위한 플로우차트,

도 11은 프리로드된 프리로드 대상 파일 중 적어도 하나를 삭제하는 방법을 설명하기 위한 플로우차트,

도 12는 도 11의 삭제 실행단계(1102단계)의 일 구현예,

도 13은 도 1과 동일한 순서로 AV 데이터 및 마크업 문서가 기록되어 있을 때 본 발명에 따른 프리로드 수행에 의한 효과를 설명하기 위한 참고도,

도 14는 도 3의 재생장치의 일부 상세도,

도 15 내지 16a, 16b, 16c, 16d, 16e, 16f는 프리로드/삭제 또는 가비지 콜렉션을 통해 메모리 관리 테이블(31)과 데이터 공간(32)을 관리하는 방법을 설명하기 위한 메모리 맵,

도 17은 제1 메모리(2)에서 AV 데이터가 로드되고 소모되는 상태도,

도 18은 프리로드 리스트 파일 및 해당 프리로드 대상 파일의 데이터 배치도(data alignment),

도 19a는 디스크의 개략도이고, 도 19b는 도 19a의 부분 상세도,

도 20은 전술한 예에 따른 제1 메모리(2)와 제2 메모리(3)의 상태도,

도 21은 본 발명의 바람직한 실시예에 따른 기록방법을 설명하기 위한 플로우차트이다.

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 프리로드 정보가 기록된 정보저장매체, 그 재생장치 및 재생방법에 관한 것으로, 보다 상세하게는 콘텐츠와 함께 마크업 문서가 기록되어 있어 디스플레이 가능한 정보저장매체, 그 재생장치 및 재생방법에 관한 것이다.

콘텐츠와 함께 마크업 문서가 기록된 DVD(이하 "인터랙티브 DVD"라 함)가 판매되고 있다. 인터랙티브 DVD에 기록된 콘텐츠는 두 가지 방법으로 재생될 수 있다. 하나는 일반 DVD와 동일하게 디스플레이되는 비디오 모드이고, 다른 하나는 마크업 문서에 의해 정의되는 표시창을 통해 디스플레이되는 인터랙티브 모드이다. 사용자에 의해 인터랙티브 모드가 선택되면 DVD 재생장치에 탑재된 웹브라우저는 인터랙티브 DVD에 기록되어 있는 마크업 문서를 디스플레이한다. 마크업 문서에 의해 정의되는 표시창에는 사용자가 선택한 콘텐츠가 표시된다.

예를 들어, 콘텐츠가 AV 데이터인 영화일 경우 마크업 문서의 표시창에는 영화가 상영되고 나머지 부분에는 대본, 줄거리, 출연배우의 사진, 등 다양한 부가정보가 표시될 수 있다. 부가정보는 이미지파일 또는 텍스트파일을 포함한다.

도 1은 AV 데이터가 기록된 인터랙티브 DVD의 개략도이다.

도 1을 참조하면, 인터랙티브 DVD의 트랙에는 AV 데이터가 MPEG 비트스트림으로 기록되어 있고 복수개의 마크업 문서가 기록되어 있다. 여기서, 마크업 문서는 마크업 문서에 삽입되는 다양한 이미지 파일, 그래픽 파일 등을 망라한 마크업 리소스(resource)를 의미한다.

도 2는 도 1의 인터랙티브 DVD를 재생하는 과정에서 발생할 수 있는 끊김현상을 설명하기 위한 참고도이다.

도 2를 참조하면, AV 데이터가 버퍼링되는 버퍼 메모리의 상태(occupancy)와 웹 리소스가 캐싱되는 캐시 메모리의 상태가 표시되어 있다. 도 1과 도 2를 참조하여 AV 데이터가 메모리에 로드되어 디스플레이되는 과정을 살펴보면, 픽업장치는 STARTUP.HTM을 탐색하고 독출하여 캐시 메모리에 로드한다. 로드된 STARTUP.HTM은 활성화된다. 이와 동시에 사용자의 선택에 따른 ① AV 데이터가 버퍼 메모리로 로드된 다음 디스플레이되기 시작한다. 이어서 ② AV 데이터가 로드되어 디스플레이된다. ② AV 데이터의 버퍼링이 완료되면 픽업장치는 ③ AV 데이터가 기록된 위치로 점프하여 버퍼링을 시작한다. 이 때, 사용자가 ④ A.HTM을 요구하면 픽업장치는 ③ AV 데이터에 대한 버퍼링을 중단하고 ④ A.HTM을 탐색하여 캐시 메모리로 로드한다. 그 동안에도 ③ AV 데이터는 계속 디스플레이되기 때문에 버퍼 메모리에 로드되어 디스플레이가 가능한 데이터는 급격히 줄어들게 된다. ④ A.HTM이 활성화되고 ③ AV 데이터의 버퍼링이 완료된 다음 이어서 ⑤ AV 데이터를 버퍼링한다. ⑤ AV 데이터의 버퍼링이 완료되면 픽업장치는 ⑥ AV 데이터가 기록된 위치로 점프한다. 이와 같은 경우, 버퍼링되어 있던 데이터가 모두 소진되는 현상이 발생할 수 있다. 즉, 종래 인터랙티브- DVD에 있어서, DVD- Video의 영상과 마크업 문서가 서로 동기되어 보여져야 할 경우(예: 특정 배우가 등장하면 그 배우에 대한 약력이 표시되는 경우) 픽업장치는 AV 데이터의 버퍼링을 중단하고 대응 마크업 문서를 탐색하여 캐싱해야 하기 때문에 영상이 갑자기 끊어지는 현상이 발생할 수 있다.

발명이 이루고자 하는 기술적 과제

따라서, 본 발명의 목적은 마크업 문서에 의해 정의된 표시창에 소정 콘텐츠를 재생하는 경우 끊김현상이 발생되지 않도록 하는 정보저장매체, 그 재생장치 및 재생방법을 제공하는 것이다.

본 발명의 다른 목적은 마크업 문서를 통해 소정 콘텐츠를 재생하는 경우 끊김현상이 발생되지 않도록, 동기되어 재생되어야 할 소정 마크업 문서가 캐시 메모리에 로드되거나 캐시 메모리로부터 삭제되도록 하는 마크업 문서가 기록된 정보저장매체, 그 재생장치 및 재생방법을 제공하는 것이다.

본 발명의 또 다른 목적은 마크업 문서에 의해 정의된 표시창에 소정 콘텐츠를 재생하는 경우 끊김현상이 발생되지 않도록 프리로드 대상 파일에 대한 속성정보를 미리 제공함으로써 보다 효율적으로 프리로드할 수 있게 해주는 정보저장매체, 그 재생장치 및 재생방법을 제공하는 것이다.

본 발명의 또 다른 목적은 콘텐츠의 재생 중 프리로드를 수행하더라도 메모리에 남아있는 데이터가 부족한 상황이 발생되지 않도록 보장하기 위한 방법을 제공하는 것이다.

본 발명의 또 다른 목적은 안정적인 프리로드 및 삭제가 수행되도록 메모리를 관리하는 방법을 제공하는 것이다.

발명의 구성 및 작용

상기 목적은 본 발명에 따라, 오디오 및/또는 비디오 데이터를 포함하는 AV 데이터; 및 디코딩되어 재생된 상기 AV 데이터를 디스플레이하고, 심리스 재생이 보장되는 AV 데이터에 대한 프리로드 대상 파일을 독출하여 메모리에 저장할 것을 명령하는 프리로드 정보가 삽입되어 있는 마크업 문서를 포함하는 것을 특징으로 하는 정보저장매체에 의해 달성된다.

상기 정보저장매체는 상기 AV 데이터에 대한 네비게이션 데이터를 더 포함하고, 상기 AV 데이터는 상기 네비게이션 데이터를 참조하여 AV 데이터 스트림으로 디코딩되는 것이 바람직하다.

상기 프리로드 대상 파일이 기록된 프리로드 리스트 파일, 및 적어도 하나의 상기 프리로드 대상 파일을 더 포함하는 것이 효과적이다.

상기 프리로드 정보는 상기 프리로드 리스트 파일의 위치정보가 기록되어 있으며 헤드 태그 내에 삽입되는 링크 태그로 구현되거나, 상기 프리로드 리스트 파일의 위치정보 및 속성을 매개변수로 갖는 API(Application Program Interface)로 구현되는 것이 바람직하다. 또한, 상기 프리로드 리스트 파일에는 상기 프리로드 대상 파일의 위치정보 및 속성이 기록되어 있다.

또한, 상기 목적은 오디오 데이터 또는 비디오 데이터를 포함하는 AV 데이터; 및 디코딩되어 AV 데이터 스트림으로 재생된 상기 AV 데이터를 디스플레이하고, 심리스 재생이 보장되는 AV 데이터에 대한 프리로드 대상 파일을 독출하여 메모리에 저장할 것을 명령하는 프리로드 정보가 삽입되어 있는 마크업 문서를 포함하는 것을 특징으로 하는 정보저장매체에 의해서도 달성된다.

상기 프리로드 정보는 상기 프리로드 리스트 파일의 위치정보 및 속성을 매개변수로 갖는 API(Application Program Interface)로 구현되며, 상기 위치정보는 상기 프리로드 리스트 파일의 경로로 표시되고, 상기 경로에는 상기 메모리, 상기 정보저장매체, 인터넷 서버 중 어느 하나를 가리키는 리소스 로케이터가 추가되어 있는 것이 바람직하다.

나아가, 상기 마크업 문서는 삭제 리스트가 기록된 삭제 리스트 파일; 및 상기 삭제 리스트에 기록된 삭제 대상 파일을 상기 메모리로부터 삭제할 것을 명령하는 삭제 정보를 더 포함하는 것이 효과적이다.

한편, 본 발명의 다른 분야에 따르면 상기 목적은 정보저장매체에 기록된 AV 데이터를 인터랙티브 모드로 재생하는 방법에 있어서, (a) 독출된 마크업 문서에 포함된 프리로드 정보를 해석하는 단계; (b) 해석된 프리로드 정보에 따른, 심리스 재생이 보장되는 AV 데이터에 대한 프리로드 대상 파일을 호출하여 캐시 메모리에 저장하는 단계; (c) 상기 AV 데이터를 독출하여 버퍼 메모리에 저장하는 단계; 및 (d) 상기 버퍼 메모리로부터 상기 AV 데이터를 독출하고, 상기 캐시 메모리로부터 상기 프리로드 대상 파일을 독출하여 인터랙티브 모드로 재생하는 단계를 포함하는 것을 특징으로 하는 재생방법에 의해서도 달성된다.

상기 (a) 단계는 (a1) 프리로드 대상 파일의 경로 및 속성을 파악하는 단계임이 바람직하며, 상기 (a1) 단계는 헤드 태그 내에 있는 링크 태그 내에 기록된 상기 프리로드 리스트 파일의 경로를 인식하는 단계임이 더욱 바람직하다.

상기 (b) 단계는 (b1) 파악된 경로로부터 상기 프리로드 대상 파일을 독출하는 단계; 및 (b2) 독출된 프리로드 대상 파일을 파악된 속성에 따라 처리하여 저장하는 단계를 포함하는 것이 특히 바람직하다.

한편, 본 발명의 다른 분야에 따르면 상기 목적은 정보저장매체에 기록된 AV 데이터를 인터랙티브 모드로 재생하는 장치에 있어서, 마크업 문서 또는 AV 데이터를 독출하는 독출부; 수신된 프리로드 파일 또는 AV 데이터를 저장하는 메모리; 상기 메모리에 저장된 AV 데이터를 디코딩하는 AV 디코더; 및 독출된 마크업 문서에 포함된 프리로드 정보를 해석하여 해석된, 심리스 재생이 보장되는 AV 데이터에 대한 프리로드 정보에 따른 프리로드 대상 파일이 먼저 상기 메모리에 저장되도록 요청하고 나서, 상기 AV 데이터가 독출되어 상기 메모리에 저장되도록 요청하고, 상기 메모리로부터 상기 프리로드 대상 파일을 독출하여 상기 AV 디코더로부터 출력된 AV 데이터와 함께 디스플레이하는 프리젠테이션 엔진을 포함하는 것을 특징으로 하는 재생장치에 의해서도 달성된다.

상기 메모리는 상기 AV 데이터를 저장하는 버퍼 메모리; 및 상기 프리로드 대상 파일을 저장하는 캐시 메모리를 포함한다.

상기 프리젠테이션 엔진은 상기 프리로드 정보에 기초하여 프리로드 대상 파일의 경로 및 속성을 파악하고 파악된 경로로부터 프리로드 대상 파일을 호출한 다음 상기 속성에 대응되도록 상기 캐시 메모리에 저장하는 것이 바람직하다.

상기 프리젠테이션 엔진은 상기 독출부 또는 인터넷 서버로 상기 프리로드 대상 파일을 요청하고, 상기 프리젠테이션 엔진은 상기 캐시 메모리의 잔여공간의 크기와 상기 프리로드 대상 파일의 크기를 비교하여 상기 캐시 메모리의 잔여공간의 크기가 더 작을 경우 에러이벤트를 발생시키며, 상기 프리젠테이션 엔진은 상기 프리로드 리스트 파일의 경로에 상기 캐시 메모리를 가리키는 리소스 로케이터가 부가되어 있는 경우 상기 캐시 메모리로 대응 프리로드 대상 파일을 요청하고 상기 캐시 메모리에 상기 프리로드 대상 파일이 존재하지 않을 경우 에러이벤트를 발생시킨다.

한편, 본 발명의 다른 분야에 따르면, 상기 목적은 프리로드를 수행하는 방법에 있어서, (a) 프리로드 대상 파일의 독출 속도를 알아내는 단계; (b) 대응 AV 데이터가 끊기지 않고 재생되도록 버퍼링이 수행되기 위한 조건을 알아내는 단계; 및 (c) 알아낸 조건에 부합하는 시점에서 프리로드를 수행하는 단계를 포함하는 것을 특징으로 하는 방법에 의해서도 달성된다.

또한, 정보저장매체에 데이터를 기록하는 방법에 있어서, (a) 프리로드 대상 파일을 기록하는 단계; (b) 기록된 프리로드 대상 파일의 독출 속도를 알아내는 단계; (c) 대응 AV 데이터가 끊기지 않고 재생되도록 버퍼링이 수행되기 위한 조건을 알아내는 단계; 및 (d) 알아낸 조건에 부합하는 시점에서 프리로드가 수행되기 위한 소스 코드를 기록하는 단계를 포함하는 것을 특징으로 하는 방법에 의해서도 달성된다.

또한, 프리로드를 위한 메모리 관리방법에 있어서, (a) 프리로드 대상 파일의 상태 정보가 포함된 메모리 관리 테이블을 기록하는 단계; 및 (b) 상기 상태 정보를 기초로 상기 프리로드 대상 파일을 삭제하는 단계를 포함하는 것을 특징으로 하는 방법에 의해서도 달성된다.

상기 프리로드를 위한 메모리 관리방법은 (c) 상기 상태 정보를 기초로 상기 프리로드 대상 파일에 대한 가비지 콜렉션을 수행하는 단계를 더 포함하는 것이 바람직하다.

상기 (b) 단계는 해당 프리로드 대상 파일에 대한 사용이 완료되면 사용 중이 아님과 삭제가능함을 상기 상태 정보로서 표시하는 단계를 포함하는 것이 바람직하다.

상기 (c) 단계는 (c1) 해당 프리로드 대상 파일이 사용 중이 아니고 삭제가능한 경우 해당 프리로드 파일을 물리적으로 삭제하는 단계; 및 (c2) 삭제된 프리로드 대상 파일이 삭제되어 존재하지 않음을 상기 상태 정보로서 표시하는 단계를 포함하는 것이 바람직하고, (c3) 메모리에 기록된 다른 프리로드 대상 파일을 재배열하는 단계를 포함하는 것이 더욱 바람직하다.

이하, 첨부된 도면을 참조하여 본 발명 바람직한 실시예를 상세히 설명한다. 본 명세서에서 「마크업 문서」는 마크업 문서는 물론 마크업 문서에 삽입되거나 링크되는 파일을 망라하는 웹 리소스(resource)를 의미한다. 나아가, 「~.HTM」는 HTML 뿐 아니라 웹브라우저에 의해 표시될 수 있는 XML, SGML 등의 마크업언어로 작성된 문서를 가리키는 의미로 사용된다.

도 3은 본 발명의 바람직한 실시예에 따른 재생장치의 블록도이다.

도 3을 참조하면, 재생장치는 본 실시예에 따른 DVD(300)에 기록된 AV 데이터를 디코딩하여 AV 데이터 스트림으로 재생한 다음 마크업 문서에 의해 정의되는 표시창을 통해 디스플레이할 수 있는 인터랙티브 모드를 지원하는 장치로서, 독출부(1), 제1 메모리(2), 제2 메모리(3), AV 디코더(4), 및 프리젠테이션 엔진(5)을 포함한다. 인터랙티브 모드에서 디스플레이되는 화면은 인터랙티브 화면으로, 마크업 화면에 AV 화면이 매립된 화면을 가리킨다. 마크업 화면은 마크업 문서가 디스플레이된 화면이고 AV 화면은 AV 데이터가 재생되어 얻어지는 화면을 말한다.

프리젠테이션 엔진(5)은 후술하는 바와 같이 링크 태그, 자바스크립트(Javascript) API(Application Program Interface) 또는 자바 애플릿(Java applet) API로 구현된 프리로드 정보 또는 자바스크립트 API 또는 자바 애플릿 API로 구현된 삭제 정보가 해석되어 실행될 수 있도록 링크 태그의 확장, 자바스크립트, 또는 자바 애플릿을 지원한다.

독출부(1)는 DVD(300)로부터 마크업 문서 또는 AV 데이터를 독출한다. 제1 메모리(2)는 버퍼 메모리로서 독출부(1)에 의해 독출된 AV 데이터를 버퍼링한다. 제2 메모리(3)는 캐시 메모리로서 수신된 프리로드 파일을 캐싱한다. AV 디코더(4)는 제1 메모리(2)에 저장된 AV 데이터를 디코딩하여 AV 데이터 스트림을 출력한다. 프리젠테이션 엔진(5)은 독출된 마크업 문서에 포함된 프리로드 정보를 해석하고, 해석된 프리로드 정보에 따라 제2 메모리(3)에 프리로드 되도록 독출부(1) 또는 인터넷 서버(도시되지 않음)로 프리로드 대상 파일을 요청한다. 또한, 프리로드 대상 파일이 AV 데이터와 동기되어 디스플레이되어야 할 경우 그 프리로드 대상 파일을 제2 메모리(3)로부터 독출하여 AV 디코더(4)로부터 출력된 AV 데이터 스트림과 함께 디스플레이되도록 한다. 또한, 삭제 정보를 해석하여 삭제 대상 파일을 제2 메모리(3)로부터 삭제한다.

본 실시예에 따른 DVD(300)에는 오디오 데이터 또는 비디오 데이터를 포함하는 AV 데이터가 기록되어 있는 한편, 프리로드 정보 및/또는 삭제 정보가 포함된 마크업 문서가 기록되어 있다. 나아가, 프리로드 리스트 파일 및/또는 삭제 리스트 파일이 더 기록될 수 있다.

프리로드 리스트 파일에는 프리로드되어야 할 파일의 명칭이 나열된 프리로드 리스트 및 각 프리로드 대상 파일을 저장하기 위해 필요한 메모리의 크기정보가 기록되어 있다. 프리로드 대상 파일은 대응 AV 데이터가 디스플레이될 경우 동기되어 재생되어야 할 마크업 문서로서 본 실시예에 따라 DVD(300)에 기록되어 있다. 다만, 프리로드 대상 파일은 인터넷을 통해 접속가능한 인터넷 서버에 저장되어 있을 수 있다.

본 발명에 따른 「프리로드 정보」는 프리로드 대상 파일을 독출하여 캐시 메모리에 저장할 것을 명령하는 정보이다. 프리로드 정보는 일 예로, 프리로드 리스트 파일의 경로 및/또는 속성이 삽입되어 있는 링크 태그로 구현될 수 있다. 이때 링크 태그는 헤드 태그 내에 삽입된다. 다른 예로, 프리로드 리스트 파일의 경로 및/또는 속성을 매개변수로 가지며 프리로드 리스트 파일을 호출하는 자바스크립트 API 또는 자바 애플릿 API로 구현된다. 또 다른 예로, 직접 프리로드 대상 파일의 경로 및/속성을 매개변수로 가지며 프리로드 대상 파일을 호출하는 자바스크립트 API 또는 자바 애플릿 API로 구현된다. 이 경우 프리로드 리스트 파일은 존재하지 않는다.

속성(attribute or property)은 일종의 MIME(Multi- Purpose Internet Mail Extensions) 헤더와 같은 역할을 하는 정보이다. 즉, 속성은 프리로드 대상 파일의 데이터 구조를 알려준다. 데이터 구조를 알면 보다 효율적으로 그 파일을 처리할 수 있다. 가령, 웹 파일인 경우 프리로드할 때 태그들을 미리 해석해두면 이후 파일분석절차를 밟지 않고서 곧바로 해당 파일을 디스플레이할 수 있다. 이미지파일인 경우 프리로드할 때 헤더 정보를 제거하고 저장할 수 있다. 이는 메모리 공간을 효율적으로 이용하게 해주며 해당 파일에 대한 탐색속도를 개선하게 해준다. audio 파일인 경우 프리로드 할 때 보다 높은 속도(rate)로 리샘플링하여 저장할 수 있다. 폰트 파일인 경우 필요한 폰트 리소스만을 추출하여 저장할 수 있다. 이처럼 프리로드 대상 파일의 속성을 파악하면 보다 효율적으로 유연하게 프리로드를 실행할 수 있다.

경로는 해당 파일이 기록된 위치를 알려준다. 프리로드 리스트 파일 및 프리로드 대상 파일의 경로에는 리소스 로케이터(Resource Locator)가 부가될 수 있다. 실제로, 웹 리소스는 DVD(300)에 기록되어 있거나 제2 메모리(3)에 캐시되어 있거나 인터넷을 통해 연결가능한 서버에 존재할 수 있다. 따라서, 리소스 로케이터는 DVD(300)를 가리키는 DVD 리소스 로케이터, 제2 메모리(3)를 가리키는 캐시 리소스 로케이터, 및 인터넷 서버를 가리키는 인터넷 리소스 로케이터로 구분된다. 리소스 로케이터들은 언급된 순서대로 다음과 같이 표시될 수 있다.

disk0://

lid://

http://

따라서, DVD(300)에 기록되어 있는 프리로드 대상 파일 A.HTM을 호출할 경우 사용되는 경로는 disk0://DVD_ENAV/A.HTM으로 표시되며, 제2 메모리(3)에 캐시되어 있는 프리로드 대상 파일 A.HTM을 호출할 경우 lid://DVD_ENAV/A.HTM로 표시되며, 인터넷 서버에 저장되어 있는 프리로드 대상 파일 A.HTM을 호출할 경우 http://www.samsung.com/DVD_ENAV/A.HTM로 표시된다. 나아가, 여러개의 DVD(300)가 로딩될 경우 DVD 각각의 리소스 로케이터는 disk0://(또는 dvd://), disk1://, disk2://, disk3://, .. 으로 표시될 수 있다.

프리로드 대상 파일의 위치를 나타내는 경로에 리소스 로케이터가 부가될 경우 리소스 로케이터가 가리키는 위치에 대상 파일이 존재하지 않으면 프리젠테이션 엔진(5)은 에러이벤트를 발생시키고 프리로드 작업을 종료시킨다. 그러나, 리소스 로케이터가 부가되어 있지 않을 경우에는 먼저 제2 메모리(3)가 탐색되고 제2 메모리(3)에 존재하지 않을 경우 DVD(300)가 탐색되도록 설정된다.

삭제 리스트 파일에는 삭제되어야 할 파일의 위치정보(파일명 및 경로)가 나열된 삭제 리스트가 기록되어 있다. 「삭제 정보」는 삭제 대상 파일을 제2 메모리(3)로부터 삭제할 것을 명령하는 정보이다. 삭제 정보는 일 예로 삭제 리스트 파일의 위치정보를 매개변수로 가지며 삭제 리스트 파일에 기록된 삭제 대상 파일을 삭제하는 자바스크립트 API 또는 자바 애플릿 API로 구현된다. 다른 예로, 직접 삭제 대상 파일의 경로 및/또는 속성을 매개변수로 가지며 삭제 대상 파일을 삭제하는 자바스크립트 API 또는 자바 애플릿 API로 구현된다. 이 경우 삭제 리스트 파일은 존재하지 않는다.

도 4a 및 4b는 본 실시예에 따른 DVD(300)의 파일 구조를 도시한 참고도이다.

도 4a를 참조하면, 루트 디렉토리에는 AV 데이터가 포함된 DVD 비디오 디렉토리 VIDEO_TS와 마크업 문서 등 인터랙티브 기능을 지원하기 위한 데이터가 기록된 DVD 인터랙티브 디렉토리 DVD_ENAV가 마련되어 있다.

VIDEO_TS에는 포함된 비디오 타이틀 전체에 대한 헤더 정보가 기록된 VIDEO_TS.IFO 및 첫 번째 비디오 타이틀에 대한 네비게이션 정보가 기록된 VTS_01_0.IFO가 기록되어 있고, 이어서 비디오 타이틀을 구성하는 AV 데이터인 VTS_01_0.VOB, VTS_01_1.VOB,... 이 기록되어 있다. 보다 상세한 구성은 DVD- Video 표준인 「DVD- Video for Read Only Memory Disc 1.0」에 개시되어 있다.

DVD_ENAV에는 인터랙티브 정보 전체에 대한 네비게이션 정보가 기록된 DVD_ENAV.IFO가 기록되어 있고, 시작 문서로 지정된 STARTUP.HTM이 기록되어 있으며, 본 실시예에 따른 프리로드 리스트 파일인 STARTUP.PLD가 마련되어 있다. 또한, 프리로드 대상 파일인 A.HTM, 또는 A.HTM에 삽입되어 표시되기 위한 그래픽 파일 A.PNG가 기록되어 있다. 기타 프리로드 대상 파일 및 이에 삽입되어 표시되기 위한 다양한 형식의 파일이 기록될 수 있다.

다만, 도 4b를 참조하면, 마크업 문서에 포함된 프리로드 정보가 직접 프리로드 대상 파일의 경로와 속성을 매개변수로 가지며 프리로드 대상 파일을 프리로드하는 API로 구현되는 경우 프리로드 리스트 파일 STARTUP.PLD는 존재하지 않는다.

도 5a 및 5b는 본 실시예에 따른 DVD(300)의 볼륨 공간(Volume Space)의 개략도이다.

도 5a를 참조하면, DVD(300)의 Volume Space에는 Volume과 파일에 대한 제어 정보가 기록된 제어정보 영역, 대응 비디오 타이틀 데이터가 기록된 DVD- Video 데이터 영역, 및 인터랙티브 모드로 재생될 수 있도록 하는 DVD- Interactive Data 영역을 포함한다.

DVD- Video 데이터 영역에는 도 4의 DVD 비디오 디렉토리 DVD_TS에 저장된 파일, 즉 VIDEO_TS.IFO, VTS_01_0.IFO, VTS_01_0.VOB, VTS_01_1.VOB,... 등이 기록되어 있다. DVD- Interactive 데이터 영역에는 도 4의 DVD 인터랙티브 디렉토리 DVD_ENAV에 저장된 파일, 즉 STARTUP.HTM, STARTUP.PLD, A.HTM 및 A.PNG이 기록되어 있다.

마찬가지로, 도 5b를 참조하면, 마크업 문서에 포함된 프리로드 정보가 직접 프리로드 대상 파일의 경로와 속성을 매개변수로 가지며 프리로드 대상 파일을 프리로드하는 API로 구현되는 경우 프리로드 리스트 파일 STARTUP.PLD는 존재하지 않는다.

도 6은 본 발명에 따른 프리로드 방식을 인터랙티브 화면(마크업 화면과 AV 화면)의 관점에서 설명하기 위한 참고도이다.

도 6을 참조하면, AV 데이터가 재생되어 얻어진 AV 화면이 도시되어 있고, 그 AV 데이터를 인터랙티브 모드로 재생하였을 때 디스플레이되는 인터랙티브 화면이 도시되어 있다. 인터랙티브 화면에는 AV 화면 및 마크업 화면으로 구성된다.

AV 데이터는 심리스(seamless) 재생이 보장되는 데이터와 그렇지 않은 데이터로 구분할 수 있다. 예를 들어, 1부, 2부, 3부로 나누어진 전쟁 영화 타이틀을 구성하는 AV 데이터가 존재할 때 1부는 전쟁 영화 타이틀을 구성하는 디폴트 부분이고 2부 및 3부는 각각 사용자의 선택에 의해 서로 다른 스토리로 전개되는 선택 부분이라고 하면 1부는 심리스 재생이 보장되는 데이터이고, 2부 및 3부는 1부의 관점에서 보면 심리스 재생이 보장되지 않는 데이터라고 부른다. 다만, 2부 또는 3부가 선택되어 재생이 되기 시작하면 2부 및 3부는 각각 심리스 재생되어야 한다. 따라서, 2부가 선택되면 2부의 입장에서 1부 및 3부는 심리스 재생이 보장되지 않는 데이터이고, 3부가 선택되면 3부의 입장에서 1부 및 2부는 심리스 재생이 보장되지 않는 데이터가 된다.

1부를 인터랙티브 모드로 재생하기 위해 필요한 마크업 문서는 STARTUP.HTM과 A.HTM이고, 2부 또는 3부를 인터랙티브 모드로 재생하기 위해 필요한 마크업 문서는 각각 OTHER1.HTM 및 OTHER2.HTM이라고 할 때, 본 발명에 따른 프리로드 정보를 기초로 프리로드되는 프리로드 대상 파일은 심리스 재생이 보장되는 AV 데이터를 인터랙티브 모드로 재생하기 위해 필요한 마크업 문서를 가리킨다. 즉, 1부가 재생될 때 프리로드 대상 파일은 STARTUP.HTM과 A.HTM이고, 2부가 재생될 때 프리로드 대상 파일은 OTHER1.HTM이며, 3부가 재생될 때 프리로드 대상 파일은 OTHER2.HTM이다.

상기와 같은 구성에 의한 본 발명에 따른 재생방법을 설명하면 다음과 같다.

도 7은 본 발명의 바람직한 실시예에 따른 재생방법을 설명하기 위한 플로우차트이다.

도 7을 참조하면, 인터랙티브 모드가 선택되면 독출부(1)는 DVD(300)에 기록되어 있는 마크업 문서로서 본 실시예에 따른 HTML 문서를 독출하고(701단계), 프리젠테이션 엔진(5)은 HTML 문서에 포함되어 있는 프리로드 정보를 해석하여 독출부(1) 또는 인터넷 서버로 프리로드를 요청한다(702단계). 이에, 프리로드 대상 파일이 캐시 메모리인 제2 메모리(3)에 저장된다(703단계).

한편, 독출부(1)는 대응 AV 데이터를 DVD(300)로부터 독출하여 버퍼 메모리인 제1 메모리(2)에 저장한다(704단계). AV 디코더(4)는 제1 메모리(1)에 저장된 AV 데이터를 디코딩하는 한편(705단계), 프리젠테이션 엔진(5)은 제2 메모리(3)로부터 프리로드 대상 파일을 독출하고 AV 디코더(4)에 의해 디코딩된 AV 데이터 스트림이 HTML 문서에 의해 정의된 표시창에 디스플레이되도록 한다(706단계).

도 8은 도 7의 프리로드 정보 해석단계(602단계)의 일 구현예이다.

도 8을 참조하면, 프리젠테이션 엔진(5)은 HTML 문서에 기록된 프리로드 리스트 파일의 경로를 인식하고(801단계), 인식된 경로로부터 프리로드 리스트 파일을 독출한다(802단계). 다음으로, 프리로드 리스트 파일에 기록된 프리로드 대상 파일을 인식한다(803단계). 여기서, 프리로드 대상 파일을 인식한다는 것은 프리로드 대상 파일의 경로, 나아가 속성을 파악하는 것을 의미한다.

도 9는 도 7의 프리로드 대상 파일의 프리로드 실행단계(703단계)의 일 구현예이다. 도 9를 참조하면, 프리젠테이션 엔진(5)은 HTML 문서의 헤드 태그 내에 있는 링크 태그 내에 기록된 프리로드 리스트 파일의 경로를 인식하여 프리로드 리스트 파일을 호출한다(901단계). 다음으로, 프리젠테이션 엔진(5)은 프리로드 대상 파일의 경로 및 속성을 변수로 갖는 프리로드 태그를 포함하는 프리로드 리스트 파일을 해석하여 프리로드를 실행한다(902단계).

도 10a는 도 7의 프리로드 대상 파일의 프리로드 실행단계(703단계)의 다른 구현예이다. 도 10a를 참조하면, 프리젠테이션 엔진(5)은 바디 태그 내에 삽입되어, 프리로드 리스트 파일의 경로를 매개변수로 갖는 API를 호출하여 프리로드 리스트 파일을 독출한다(1001a단계). 다음으로, 프리젠테이션 엔진(5)은 프리로드 대상 파일의 경로 및 속성을 변수로 갖는 프리로드 태그를 포함하는 프리로드 리스트 파일을 해석하여 프리로드를 실행한다(1001b단계).

도 10b는 도 7의 프리로드 대상 파일의 프리로드 실행단계(703단계)의 또 다른 구현예이다. 도 10b를 참조하면, 프리젠테이션 엔진(5)은 바디 태그 내에 삽입되어, 직접 프리로드 대상 파일의 경로 및 속성을 매개변수로 갖는 API를 호출하여 프리로드 대상 파일을 메모리에 저장한다(1001b단계). 이때, 프리로드 대상 파일의 속성을 파악할 수 있으므로 프리젠테이션 엔진(5)은 속성에 따라 프리로드 대상 파일을 처리한 다음 메모리에 저장할 수 있다.

도 11은 프리로드 리스트 파일에 프리로드 대상 파일의 크기가 포함되어 있을 경우 프리로드하는 방법을 설명하기 위한 플로우차트이다.

도 11을 참조하면, 인터랙티브 모드가 선택되면 독출부(1)는 DVD(300)에 기록되어 있는 마크업 문서로서 본 실시예에 따른 HTML 문서를 독출하고, 프리젠테이션 엔진(5)은 HTML 문서에 포함되어 있는 프리로드 정보를 해석하며 이에 독출부(1)는 프리로드 리스트 파일을 독출한다(1101단계). 프리젠테이션 엔진(5)은 프리로드 리스트 파일을 해석한다(1102단계). 프리젠테이션 엔진(5)은 프리로드 대상 파일의 크기를 인식하고 이와 캐시 메모리의 잔여용량을 비교한다(1103단계). 프리로드 대상 파일의 크기가 캐시 메모리의 잔여용량보다 작은 경우 프리로드를 실행한다(1104단계). 프리로드 대상 파일의 크기가 캐시 메모리의 잔여용량보다 큰 경우 에러이벤트를 발생시키고 그대로 종료한다(1105단계).

도 12는 메모리에 저장된 프리로드 대상 파일 중 적어도 하나를 삭제하는 방법을 설명하기 위한 플로우차트이다.

도 12를 참조하면, 프리젠테이션 엔진(5)은 HTML 문서에 포함된 삭제정보를 해석하고(1201단계), 삭제 리스트 파일로부터 인식된 삭제 대상 파일을 캐시 메모리인 제2 메모리(3)로부터 삭제한다(1202단계). 다만, 후술하는 소스코드에서 확인되듯이 본 실시예에서의 프리로드 리스트 파일과 삭제 리스트 파일은 동일한 하나의 파일, 즉 STARUP.PLD로 구현하였다. 물론 프리로드 대상 파일의 리스트와 삭제 대상 파일의 리스트가 각각 분리되어 기록된 두 개의 파일로도 구현할 수 있다.

도 13은 도 12의 삭제 실행단계(1202단계)의 일 구현예이다.

도 13을 참조하면, 삭제 리스트 파일의 경로를 매개변수로 가지고 삭제 리스트 파일에 기록된 삭제 대상 파일을 캐시 메모리인 제2 메모리(3)로부터 삭제하는 API에 의해 실행되도록 한다(1301단계). 여기서, 「삭제」의 의미는 물리적으로 해당 데이터를 지우는 가비지 컬렉션 뿐 아니라, 물리적으로 데이터는 그대로 둔 상태에서 플래그 등을 사용하여 해당 데이터가 삭제가능함을 알려거나 여기에 다른 데이터를 기록할 수 있음을 알리는 것을 포함한다.

한편, 전술한 STARTUP.HTM 및 STARTUP.PLD의 소스코드는 다음과 같이 구성될 수 있다.

※ STARTUP.HTM 예 1

[표 1]

```
< ?xml version=" 1.0" encoding=" UTF- 8" ?> < !DOCTYPE html PUBLIC - //DVD/DTD XHTML DVD- HTML1.0//EN" "
http://www.dvdforum.org/enav/dvdhtml- 1- 0.dtd" > < html> < head> < title> STARTUP PAGE< /title> < link rel="
preload" src=" dvd://dvd_enav\startup.pld" OnError=" err_preload()" OnAbort=" err_preload()" > < !- - if preloading i
s failed, call err_preload - - > < script language=" ecmaScript" > < !- - function html_discard(){ navigator.Discard(" d
vd://dvd_enav\startup.htm" ,0);}function err_preload(){ navigator.Discard(" " ,2); if (!navigator.Preload(" dvd://dvd_en
av\startup.pld" ,1)) { alert(" insufficient memory. it will change interactive mode to raw mode." ) DvdVideo.SetVideoM
ode() }}< /script> < /head> < body bgcolor=# ffffff OnUnload=" html_discard()" > < !- - if document unload, call ht
ml_discard - - > < object height=" 50%" width=" 60%" data=" dvd:" > < script language=" ecmaScript" > < !- - DvdV
ideo.Play()< /script> < a href=" lid:\\dvd_enav\A.htm" > click to preloaded A.HTM< /a> < /body> < /html>
```

위의 소스코드에는 헤드 태그 내에 삽입된 링크 태그를 통해 구현된 프리로드 정보가 삽입되어 있다. 또한, 자바스크립트 API에 의해 구현된 삭제 정보가 삽입되어 있다.

※ STARTUP.HTM의 예 2

[표 2]

```
< ?xml version=" 1.0" encoding=" UTF- 8" ?> < !DOCTYPE html PUBLIC - //DVD/DTD XHTML DVD- HTML1.0//EN" "
http://www.dvdforum.org/enav/dvdhtml- 1- 0.dtd" > < html> < head> < title> STARTUP PAGE< /title> < script lang
uage=" ecmaScript" > < !- - function html_discard(){ navigator.Discard(" dvd://dvd_enav\startup.htm" ," text/xml" );}f
unction err_preload(){ navigator.Discard(" " ," "); if (!navigator.Preload(" dvd://dvd_enav\startup.pld" ," text/preload&
amp;quot;)) { alert(" insufficient memory. it will change interactive mode to raw mode." ); DvdVideo.SetVideoMode()
; }}< /script> < /head> < body bgcolor=# ffffff OnUnload=" html_discard()" > < !- - if document unload, call html_di
scard - - > < object height=" 50%" width=" 60%" data=" dvd:"> < script language=" ecmaScript" > < !- - if (!navig
ator.Preload(" dvd://dvd_enav\startup.pld" ," text/preload& amp;quot;)) { err_preload(); } DvdVideo.Play();< /script>
< a href=" lid:\\dvd_enav\A.htm" > click to preloaded A.HTM< /a> < /body> < /html>
```

위의 소스코드에는 자바스크립트 API로 구현된 삭제 정보 및 프리로드 정보가 삽입되어 있다.

※ STARTUP.PLD의 예 1

[표 3]

```
< ?xml version=" 1.0" encoding=" UTF- 8" ?> < !DOCTYPE PRELOAD PUBLIC " - \\DVD\\DTD DVD Preload List 1.0\\E
N" " http://www.dvdforum.org/enav/dvd- preload- list.dtd" - - > < preload cachesize=" 128KB" > < filedef type=" text
/xml" src=" dvd://dvd_enav//A.htm" /> < filedef type=" image/png" src=" dvd://dvd_enav//A.png" /> < /preload>
```

위의 소스코드는 XML 문서로서 프리로드 대상 파일의 크기 및 프리로드 대상 파일의 경로 및 속성이 기록되어 있다.

한편, 전술한 소스코드에서 사용된 프리로드/삭제(Preload/Discard)를 위한 API에 대해 보다 구체적으로 설명하면 다음과 같다.

1. navigator.Preload(URL,flag)

지시하는 프리로드 대상 파일들을 제2 메모리(3)로 미리 읽어오는 API이다. 사용되는 매개변수는 프리로드 리스트 파일 및 프리로드 대상 파일의 위치정보를 나타낸다.

*108URL = : 프리로드 리스트 파일의 경로 또는 프리로드 대상 파일의 경로

flag = : 프리로드 리스트 파일인 경우 1, 프리로드 대상 파일인 경우 0

반환값:

*111프리로드에 성공하면 true를 반환하고 실패하면 false를 반환한다.

예:

navigator.Preload(" http://www.holeywood.com/tom.pld" ,1) // 인터넷에 있는

// " http://www.holeywood.com/tom.pld" 의 프리로드 리스트 파일을 읽어 지시하는 프리로드 대상 파일들을

// 캐시 메모리로 미리 읽어 와라.

2. navigator.Preload(URL,resType)

지시하는 프리로드 대상 파일들을 제2 메모리(3)로 미리 읽어오는 API이다. 사용되는 매개변수는 프리로드 리스트 파일, 프리로드 대상 파일의 위치정보, 나아가 프리로드 대상 파일의 속성을 나타낼 수 있다.

URL = : 프리로드 리스트 파일의 경로 또는 프리로드 대상 파일의 경로

resType = : 대상 파일의 속성

반환값:

프리로드에 성공하면 true를 반환하고 실패하면 false를 반환한다.

예:

navigator.Preload(" dvd://dvd_enav/a.htm" , " text/xml") // dvd에 있는 " dvd://dvd_enav/a.htm" 의 프리로드 대상 파일을 읽어와라. 그 파일은 텍스트 파일로서 xml파일이다.

navigator.Preload(" http://www.holeywood.com/tom.htm" , " text/html") // 인터넷에 있는 " http://www.holeywood.com/tom.html" 의 파일을 읽어와라. 그 파일은 텍스트 파일로서 xml파일이다.

3. navigator.Discard(URL,flag)

지시하는 삭제 대상 파일들을 제2 메모리(3)에서 삭제하는 API이다. 매개변수는 삭제 리스트 파일 및 삭제 대상 파일의 위치정보를 나타낸다.

URL = : 삭제 리스트 파일의 경로 또는 삭제 대상 파일의 경로

flag = : 삭제 리스트 파일인 경우 1, 삭제 대상 파일인 경우 0, 캐시 메모리에 로드된 내용을 전부 다 삭제하려면 2

반환값:

삭제에 성공하면 true를 반환하고 실패하면 false를 반환한다.

예:

navigator.Discard(' http://www.holeywood.com/tom.htm' ,0) // 인터넷에 있는

// 'http://www.hollywood.com/tom.htm' 의 지시하는 삭제 대상 파일이 캐시 메모리에 있으면 이를 삭제한다.

4. navigator.Discard(URL,resType)

지시하는 삭제 대상 파일들을 제2 메모리(3)에서 삭제하는 API이다. 매개변수는 삭제 리스트 파일, 삭제 대상 파일의 위치정보, 및 삭제 대상파일의 속성을 나타낸다.

URL = : 삭제 리스트 파일의 경로 또는 삭제 대상 파일의 경로

resType = : 대상 파일의 속성

반환값:

삭제에 성공하면 true를 반환하고 실패하면 false를 반환한다.

예:

navigator.Discard(" dvd://dvd_enav/a.htm" , " text/xml") // dvd에 있는 " dvd://dvd_enav/a.htm" 의 삭제 대상 파일이 캐시 메모리에 있으면 이를 삭제한다. 그 파일은 text파일로서 xml파일이다.

navigator.Discard(" dvd://dvd_enav/a.pld" , " application/preload") // dvd에 있는 " dvd://dvd_enav/a.pld" 의 삭제 대상 파일이 캐시 메모리에 있으면 이를 삭제한다. 그 파일은 삭제 대상 리스트 파일이다.

navigator.Discard(" http://www.hollywood.com/tom.htm" ," text/xml& amp;quot;) // 인터넷에 있는 " http://www.hollywood.com/tom.htm" 의 지시하는 삭제 대상 파일이 캐시 메모리에 있으면 이를 삭제한다. 그 파일은 text파일로서 xml파일이다.

한편, 전술한 실시예에서는 자바스크립트에 의해 구현된 API에 대해서만 설명하였으나 자바 애플릿 등에 의해서도 마찬가지로 결과를 얻을 수 있다.

도 14는 도 1과 동일한 순서로 AV 데이터 및 HTML 문서가 기록되어 있을 때 본 발명에 따른 프리로드 수행에 의한 효과를 설명하기 위한 참고도이다.

도 14에는 MPEG 코딩된 AV 데이터가 버퍼링되는 제1 메모리(2)의 상태(occupancy)와 웹 리소스가 캐싱되는 제2 메모리(3)의 상태가 표시되어 있다. 도 1과 도 14를 참조하여 AV 데이터가 로드되어 디스플레이되는 과정을 살펴보면, 독출부(1)는 STARTUP.HTM을 탐색하여 독출하고 프리젠테이션 엔진(5)은 STARTUP.HTM에 삽입되어 있는 프리로드 정보를 해석하여 ④ A.htm의 프리로드가 수행되도록 한다. 이에 ④ A.HTM이 제2 메모리(3)에 프리로드된다. 한편, 로드된 STARTUP.HTM은 활성화된다. 이와 동시에 사용자의 선택에 따른 ① AV 데이터가 제1 메모리(2)로 로드된 다음 디스플레이되기 시작한다. 이어서 ② AV 데이터가 로드되어 디스플레이된다. ② AV 데이터의 버퍼링이 완료되면 독출부(1)는 ③ AV 데이터를 찾아 접근하여 버퍼링을 시작한다. 이 때, 사용자가 ④ A.HTM을 요구하면 프리젠테이션 엔진(5)은 제2 메모리(3)에 프리로드되어 있는 ④ A.HTM을 독출하여 디스플레이한다. 즉, 독출부(1)가 ③ AV 데이터에 대한 버퍼링을 중단하고 ④ A.HTM을 DVD(300)로부터 탐색하여 제2 메모리(3)로 로드하는 작업을 수행하지 않아도 된다. 따라서, 독출부(1)는 버퍼링을 중단없이 수행할 수 있다. 다만, 독출부(1)가 ⑤ AV 데이터의 버퍼링을 마무리하고 ⑥ AV 데이터로 접근하는 과정에서 제1 메모리(2)에 버퍼링되어 있던 데이터가 줄어들 수 있으나 이미 버퍼링해놓은 데이터의 양이 충분하기 때문에 데이터 부족현상은 발생되지 않는다. 즉, 인터랙티브 모드를 지원하는 DVD에 있어서 DVD- Video의 영상과 HTML 문서가 서로 동기되어 보여져야 할 경우에도(예: 특정 배우가 등장하면 그 배우에 대한 약력이 표시되는 경우) 대응 HTML 문서가 이미 제2 메모리(3)에 프리로드되어 있기 때문에 독출부(1)는 AV 데이터의 버퍼링을 중단하고 대응 HTML 문서를 탐색하여 캐싱해야 할 필요가 없다.

이하에서는 안정적인 프리로드/삭제를 수행할 수 있도록 제2 메모리(3)를 관리하는 방법 및 프리로드를 수행함에 있어 제1 메모리(2)에 남아있는 콘텐츠가 소진되는 상황이 발생되지 않도록 하기 위한 보다 구체적인 방법을 설명한다.

도 15는 도 3의 재생장치의 일부 상세도이다.

도 15를 참조하면, 제2 메모리(3)에는 메모리 관리 테이블 영역(31)과 데이터 영역(32)이 마련되어 있다. 메모리 관리 테이블(31)은 데이터 영역(32)에 기록된 데이터를 기록/삭제 등 관리하기 위한 정보가 기록되는 테이블이다. 데이터 영역에는 프리로드된 마크업 문서 등이 기록된다. 한편, 프리젠테이션 엔진(5)은 자바스크립트 해석엔진(51), 및 실행 모듈(52)로 구분할 수 있다. 실행 모듈(52)은 프리로드/삭제 모듈(521) 및 가비지 콜렉션(garbage collection) 모듈(522)로 구성된다. 가비지 콜렉션(garbage collection) 모듈(522)에 대해서는 후술한다.

본 실시예에서 자바스크립트 해석엔진(51)은 자바스크립트로 작성된 API를 호출한다. 프리로드/삭제 모듈(521) 또는 가비지 콜렉션 모듈(522)은 프리로드/삭제 또는 가비지 콜렉션을 수행한다.

도 16 내지 17은 프리로드/삭제 또는 가비지 콜렉션을 통해 메모리 관리 테이블(31)과 데이터 공간(32)을 관리하는 방법을 설명하기 위한 메모리 맵이다.

도 16을 참조하면, 메모리 관리 테이블(31)은 프리로드 대상 파일의 상태 정보, 프리로드 대상 파일이 저장된 경로, 데이터 포인터 및 데이터 크기 정보를 포함한다. "in use"는 현재 사용 중인지 아닌지를 나타내고, "discordable"은 삭제가능 여부를 나타내며, "URI"는 경로 정보를, "data pointer"는 데이터의 시작주소를 나타내고, "size"는 데이터의 크기를 나타낸다. 현재, 데이터 영역(32)에는 a.htm, c.htm 및 b.htm이 로드되어 있음을 알 수 있다.

도 17a를 참조하면, 현재 a.htm을 사용 중이며 b.htm, c.htm 및 d.htm을 프리로드하였다면 a.htm은 현재 사용 중이므로 a.htm의 "in use" 값은 1이고 나머지 파일은 열려 있지 않으므로 그 "in use" 값은 모두 0이다.

도 17b를 참조하면, a.htm의 사용이 종료되고 b.htm가 사용 중이므로 a.htm의 "in use" 값은 0으로, b.htm의 "in use" 값은 1로 바뀐다.

도 17c를 참조하면, a.htm에 대해 가비지 콜렉션이 수행된다. 가비지 콜렉션이 수행되면 데이터 영역(32)에서 a.htm이 삭제되고 b.htm, c.htm 및 d.htm이 재배열된다. a.htm의 "data pointer" 값은 -1로 표시된다. 이는 데이터 영역(32)에 해당 파일이 존재하지 않음을 뜻한다.

도 17d를 참조하면, f.htm가 사용 중임을 나타낸다. f.htm은 메모리 관리 테이블(31)에서 도 16에서 a.htm의 자리에 배치되며 "data pointer" 값은 실제로 f.htm이 기록된 시작주소를 가리킨다.

도 17e를 참조하면, b.htm, c.htm, d.htm에 대한 삭제가 수행된다. 이에 b.htm, c.htm, d.htm의 "discordable" 값은 모두 1로 바뀐다.

도 17f를 참조하면, b.htm, c.htm, d.htm에 대한 가비지 콜렉션이 수행된다. 이에, b.htm, c.htm, d.htm의 "data pointer" 값은 모두 -1이 된다. 데이터 영역(32)에 기록된 b.htm, c.htm, d.htm은 모두 삭제되고 남아 있는 f.htm이 재배열된다.

이처럼, 프리로드/삭제 및 가비지 콜렉션을 통해 제2 메모리(3)가 적절히 관리될 수 있다.

도 18은 제1 메모리(2)에서 AV 데이터가 로드되고 소모되는 상태도이다.

도 18을 참조하면, 구간 a에서 앵글 블록 등으로 점프가 수행되는 동안에는 AV 데이터가 채워지지 않고 소모되기만 하

므로 속도 V_o 로 데이터가 감소됨을 표시하고 있다. 앵글 블록이란 동일한 장면을 다양한 앵글에서 촬영한 데이터가 기록된 블록을 의미한다. 하나의 앵글이 선택되어 재생되고 나머지 앵글에서 촬영한 데이터는 건너뛰게 되므로 앵글 블록에서는 필연적으로 점프가 일어난다. 점프가 완료되고 AV 데이터를 읽어들이면 다시 데이터가 채워진다. 구간 b에 기록된 바와 같이, AV 데이터를 속도 V_r 로 계속 읽어들이고 동시에 속도 V_o 로 계속 소모한다면 실제로 AV 데이터는 속도 $V_r - V_o$ 로 채워진다. 구간 c에서 마크업 문서의 프리로드가 수행되면 독출부(1)는 AV 데이터의 독출을 중지하고 해당 마크업 문서를 프리로드해야 하므로 속도 V_o 로 데이터가 감소된다. 구간 d에서는 다시 AV 데이터가 채워지므로 구간 b에서와 마찬가지로 AV 데이터가 속도 $V_r - V_o$ 로 채워진다. 한편, 점선은 채워져 있어야 하는 AV 데이터의 최소량을 표시한 것이다.

이를 바탕으로, 본 발명에 따른 프리로드가 수행되면서도 제1 메모리(2)의 버퍼링 기능이 제대로 발휘되기 위해서는(제1 메모리의 데이터 부족현상이 발생되지 않도록 하기 위해서는) 프리로드가 수행되는 구간 c에서 감소되는 데이터 양보다 남아 있는 데이터 양이 더 커야 함을 알 수 있다.

한편, 심리스(seamless) 재생을 보장하기 위해 독출부(1)는 프리로드 대상 파일을 가능한 한 연속적으로 읽어들이어야 한다. 따라서, 프리로드 대상 파일, 즉 PLD 파일은 해당 콘텐츠를 연속적으로 읽어들이 수 있도록 재생장치의 파일 시스템은 도 19에 도시된 바와 같이 하나의 PLD 파일이 다른 PLD 파일을 네스팅(nesting) 하는 구조로 데이터 정렬(data alignment) 하는 것이 효과적이다.

도 20a 및 20b는 도 18의 PLD 파일이 기록된 디스크(300)의 개략도이다.

도 20a 및 20b를 참조하면, 독출부(1)는 제2 메모리(3)에 AV 데이터 VTS_01_1.VOB를 버퍼링하다가 프리로드 리스트 파일 A.PLD에 지정된 프리로드 대상 파일 A.HTM, A1.JPG, 및 A2.JPG를 프리로드하고 나아가 프리로드 리스트 파일 B.PLD까지 프리로드한 다음 다시 VTS_01_1.VOB를 버퍼링한다.

여기서, T_j 는 프리로드 대상 파일에 대한 액세스 시간을 의미한다. 즉, $T_j = T_{j1} + T_{j2}$ 는 VTS0_1.VOB에서 프리로드 대상 파일 A1.JPG로 점프하는데 걸리는 시간 T_{j1} 과 B.PLD파일을 프리로드한 다음 다시 VTS_01_1.VOB로 점프하는데 걸리는 시간 T_{j2} 의 합이 된다. De 는 프리로드 대상 파일의 데이터 크기를 나타낸다. 즉, $De = De_1 + De_2 + De_3$ 은 A.HTM의 데이터 크기 De_1 와, A1.JPG의 데이터 크기 De_2 및 A2.JPG의 데이터 크기 De_3 의 합이 된다. T_k 는 프리로드 대상 파일을 읽어들이기 때 내부 점프 시간을 나타낸다. 즉, $T_k = T_{k1} + T_{k2}$ 로서 T_{k1} 은 A1.JPG에서 A2.JPG로 점프하는데 소요되는 시간을, T_{k2} 는 A2.JPG에서 B.PLD로 점프하는데 소요되는 시간을 가리킨다.

한편, 디스크에 기록된 AV 데이터의 액세스 거리(access distance)와 액세스 시간과의 관계는 다음과 같이 설정될 수 있다. N은 섹터 개수를 말한다.

[표 4]

access distance(sector)	0 to 210	to 5000	to 10,000	to 15,000	to 20,000	over 20,000
access time(msec)	$1.4 \times N$	310	360	390	410	1500

본 발명에 따라, 제1 메모리(2)에서 AV 데이터의 부족현상을 방지하기 위한 조건을 수식으로 표현하면 다음과 같다.

수학식 1

$$Vo \times Tp < (Vr - Vo) \times (tN_ecc \times 2048 \times 8 \times 16 / Vr) - (Vo \times Ta) - Bs$$

Vo: 제1 메모리(2)에서 AV 데이터가 빠져나가는 속도 또는

AV 디코더(4)의 데이터 소모 속도,

Tp: 구간 c에서 프리로드를 수행하는데 걸린 시간,

Vr: 디스크(300)로부터 데이터를 읽어들이는 속도

tN_ecc: 프리로드 대상 파일을 읽어들이기 이전에 읽어들여야 할 전체 ECC 블록의 갯수

Ta: 앵글 블록에서 점프하는데 걸리는 총 시간

Bs: 제1 메모리(2)에 확보되어야 할 최소한의 데이터 크기(여기서는 DVD- Video 규격 1.0에서 지정된 메모리 크기 221 sectors)

[수학식 1]에서 $Vo \times Tp$ 는 구간 c에서 프리로드 수행으로 인해 감소되는 데이터 양을 나타낸다. 한편, $tN_ecc \times 2048 \times 8 \times 16$ 은 구간 b에서 읽어들이는 데이터의 길이(섹터의 수)이고 Vr은 데이터의 독출 속도이므로 $tN_ecc \times 2048 \times 8 \times 16 / Vr$ 은 구간 b의 시간이 된다. 즉, $(Vr - Vo) \times (tN_ecc \times 2048 \times 8 \times 16 / Vr)$ 는 구간 b에서 증가되는 데이터 양을 나타낸다. 한편, $Vo \times Ta$ 는 구간 a에서 앵글 블록 점프 등으로 인해 감소되는 데이터 양을 나타낸다. Bs는 확보되어야 하는 최소 데이터 양을 나타낸다. 한편, Tp는 $2 \times Tj + De / Vr + Tk$ 를 의미한다. Tj, De, Vr 및 Tk의 의미는 전술한 바 있다.

도 18의 프리로드 리스트 파일 A.PLD, B.PLD, C.PLD에 지정된 프리로드 대상 파일들을 프리로드하는데 소요되는 시간 Tp를 계산하였더니 다음과 같은 결과가 나왔다.

$Vr = 22Mbps$,

A.PLD의 대상 파일들($De = 1611KB$): $Tp = 3600msec (= 1500msec \times 2 + 1611KB \times 8 / 22000000 + 0)$

B.PLD의 대상 파일들($De = 2685KB$): $Tp = 4000msec (= 1500msec \times 2 + 2685KB \times 8 / 22000000 + 0)$

C.PLD의 대상 파일들($De = 269KB$): $Tp = 3100msec (= 1500msec \times 2 + 269KB \times 8 / 22000000 + 0)$

VTS_01_1.VOB의 재생 중에 A.PLD의 대상 파일, B.PLD의 대상 파일, C.PLD의 대상 파일을 순서대로 열어야 한다면 다음 값을 계산해야 한다.

A.PLD의 대상 파일들을 읽어들이기 전 VTS_01_1.VOB의 버퍼링 구간에서의 Vo 값 = Va

B.PLD의 대상 파일들을 읽어들이기 전 VTS_01_1.VOB의 버퍼링 구간에서의 Vo 값 = Vb

C.PLD의 대상 파일들을 읽어들이기 전 VTS_01_1.VOB의 버퍼링 구간에서의 Vo 값 = Vc

나아가, 각 구간마다 앵글 블록과 같은 점프가 없고 Bs가 221 sectors(= 14 ECC 블록) 이라고 가정하고 각 구간별 ECC 블록의 개수와 Va, Vb, Vc를 관련지어 계산하면 AV 화면이 끊기지 않고 A.PLD, B.PLD, C.PLD의 대상 파일들을 읽어들이 수 있는 위치를 찾아낼 수 있다.

가령, $V_a = 8\text{Mbps}$, $V_b = 6\text{Mbps}$, $V_c = 4\text{Mbps}$ 로 가정하면 [수학식 1]을 사용하여 t_{N_ecc} 를 구할 수 있다. 구해보면,

A.PLD의 대상 파일들을 읽어들이기 전에 최소 187개의 ECC 블록을 읽어들여야 하며,

B.PLD의 대상 파일들을 읽어들이기 전에 최소 140개의 ECC 블록을 읽어들여야 하며,

C.PLD의 대상 파일들을 읽어들이기 전에 최소 72개의 ECC 블록을 읽어들여야 한다.

도 21은 전술한 예에 따른 제1 메모리(2)와 제2 메모리(3)의 상태도이다.

도 21을 참조하면, STARTUP.HTM이 활성화되었을 때에는 제1 메모리(2)의 데이터 양은 $V_r - V_a$ 의 속도로 증가한다. A.PLD의 대상 파일들에 대한 프리로드를 수행하면 제1 메모리(2)의 데이터 양은 V_a 의 속도로 감소한다. A.PLD의 대상 파일들에 대한 프리로드가 완료되고 그 중 A.HTM이 활성화되면 데이터 양은 $V_r - V_b$ 의 속도로 증가한다. 다시, B.PLD의 대상 파일들에 대한 프리로드가 수행되면 제1 메모리(2)의 데이터 양은 V_b 의 속도로 감소한다. B.PLD의 대상 파일들에 대한 프리로드가 완료되고 그 중 B.HTM이 활성화되면 데이터 양은 $V_r - V_c$ 의 속도로 증가한다. 한편, 제2 메모리(3)의 프리로드가 완료된 지점에서 데이터 양이 급격히 감소한 이유는 이전에 프리로드한 대상 파일의 삭제가 요구되어 가비지 콜렉션이 수행되었기 때문이다.

도 22는 본 발명의 바람직한 실시예에 따른 기록방법을 설명하기 위한 플로우차트이다.

도 22를 참조하면, 콘텐츠 제작자는 프리로드 대상 파일의 독출 속도를 알아낸다(2201단계). 대응 AV 데이터가 끊기지 않고 재생되도록 제1 메모리(2)에 버퍼링이 수행되기 위한 조건을 알아낸다(2202단계). 그 조건은 상세히 전술한 바와 같다. 알아낸 조건에 부합하는 시점에서 프리로드가 수행되도록 프리로드를 위한 소스 코드를 기록한다(2203단계). 즉, 해당 API가 호출되어 프리로드를 수행하는 시점을 최소한의 AV 데이터가 제1 메모리(2)에 버퍼링된 이후로 맞추어 기록함으로써 AV 화면의 끊김현상없이 프리로드가 수행될 수 있게 된다.

발명의 효과

전술한 바와 같이, 본 발명에 따르면 DVD 등 정보저장매체에 기록된 AV 데이터를 재생하여 마크업 문서를 통해 디스플레이하는 경우 동영상 화면의 끊김현상이 발생되지 않도록 하는 마크업 문서가 기록된 정보저장매체, 그 재생장치 및 재생방법이 제공된다. 나아가, 프리로드 대상 파일 및 삭제 대상 파일의 속성까지 알 수 있으므로 보다 효율적인 프리로드 및 삭제가 가능해진다.

(57) 청구의 범위

청구항 1.

오디오 및/또는 비디오 데이터를 포함하는 AV 데이터; 및

디코딩되어 재생된 상기 AV 데이터를 디스플레이하고, 심리스 재생이 보장되는 AV 데이터에 대한 프리로드 대상 파일을 독출하여 메모리에 저장할 것을 명령하는 프리로드 정보가 삽입되어 있는 마크업 문서를 포함하는 것을 특징으로 하는 정보저장매체.

청구항 2.

제1항에 있어서,

상기 AV 데이터에 대한 네비게이션 데이터를 더 포함하고,

상기 AV 데이터는 상기 네비게이션 데이터를 참조하여 AV 데이터 스트림으로 디코딩되는 것을 특징으로 하는 정보저장매체.

청구항 3.

제1항 또는 제2항에 있어서,

적어도 하나의 상기 프리로드 대상 파일을 더 포함하는 것을 특징으로 하는 정보저장매체.

청구항 4.

제1항에 있어서,

상기 프리로드 대상 파일이 기록된 프리로드 리스트 파일을 더 포함하며,

상기 프리로드 정보는 상기 프리로드 리스트 파일의 위치정보가 기록되어 있는 링크 태그로 구현되는 것을 특징으로 하는 정보저장매체.

청구항 5.

제4항에 있어서,

상기 링크 태그는 헤드 태그 내에 삽입되는 것을 특징으로 하는 정보저장매체.

청구항 6.

제4항 또는 제5항에 있어서,

상기 위치정보는 상기 프리로드 리스트 파일의 경로로 표시되며, 상기 경로에는 상기 메모리, 상기 정보저장매체, 인터넷 서버 중 어느 하나를 가리키는 리소스 로케이터가 부가되어 있는 것을 특징으로 하는 정보저장매체.

청구항 7.

제1항에 있어서,

상기 프리로드 정보는 상기 프리로드 리스트 파일의 위치정보를 매개변수로 갖는 API(Application Program Interface)로 구현되는 것을 특징으로 하는 정보저장매체.

청구항 8.

제7항에 있어서,

상기 API는 자바스크립트 API 또는 자바 애플릿 API임을 특징으로 하는 정보저장매체.

청구항 9.

제7항 또는 제8항에 있어서,

상기 위치정보는 상기 프리로드 리스트 파일의 경로로 표시되며, 상기 경로에는 상기 메모리, 상기 정보저장매체, 인터넷 서버 중 어느 하나를 가리키는 리소스 로케이터가 부가되어 있는 것을 특징으로 하는 정보저장매체.

청구항 10.

제1항에 있어서,

상기 프리로드 리스트 파일은 상기 프리로드 대상 파일의 위치정보 및 속성을 포함하는 것을 특징으로 하는 정보저장매체.

청구항 11.

제10항에 있어서,

상기 프리로드 리스트 파일에는 상기 프리로드 대상 파일을 저장하기 위해 필요한 메모리의 크기정보가 더 기록되어 있는 것을 특징으로 하는 정보저장매체.

청구항 12.

오디오 데이터 또는 비디오 데이터를 포함하는 AV 데이터; 및

디코딩되어 AV 데이터 스트림으로 재생된 상기 AV 데이터를 디스플레이하고, 프리로드 대상 파일을 독출하여 메모리에 저장할 것을 명령하는 프리로드 정보가 삽입되어 있는 마크업 문서를 포함하는 것을 특징으로 하는 정보저장매체.

청구항 13.

제12항에 있어서,

상기 AV 데이터에 대한 네비게이션 데이터를 더 포함하고,

상기 AV 데이터 스트림은 상기 네비게이션 데이터를 참조하여 디코딩된 것임을 특징으로 하는 정보저장매체.

청구항 14.

제12항 또는 제13항에 있어서,

적어도 하나의 상기 프리로드 대상 파일을 더 포함하는 것을 특징으로 하는 정보저장매체.

청구항 15.

제14항에 있어서,

상기 프리로드 정보는 상기 프리로드 리스트 파일의 위치정보를 매개변수로 갖는 API(Application Program Interface)로 구현되는 것을 특징으로 하는 정보저장매체.

청구항 16.

제15항에 있어서,

상기 위치정보는 상기 프리로드 리스트 파일의 경로로 표시되며, 상기 경로에는 상기 메모리, 상기 정보저장매체, 인터넷 서버 중 어느 하나를 가리키는 리소스 로케이터가 부가되어 있는 것을 특징으로 하는 정보저장매체.

청구항 17.

제15항에 있어서,

상기 프리로드 정보는 상기 프리로드 대상 파일의 위치정보 및 속성을 매개변수로 갖는 API로 구현되는 것을 특징으로 하는 정보저장매체.

청구항 18.

제1항에 있어서,

상기 마크업 문서는 삭제 리스트가 기록된 삭제 리스트 파일; 및

상기 삭제 리스트에 기록된 삭제 대상 파일을 상기 메모리로부터 삭제할 것을 명령하는 삭제 정보를 더 포함하는 것을 특징으로 하는 정보저장매체.

청구항 19.

제18항에 있어서,

상기 삭제 정보는 상기 삭제 리스트 파일의 위치정보를 매개변수로 갖는 API(Application Program Interface)로 구현되는 것을 특징으로 하는 정보저장매체.

청구항 20.

제12항에 있어서,

상기 마크업 문서는 삭제 대상 파일을 상기 메모리로부터 삭제할 것을 명령하는 삭제 정보를 더 포함하는 것을 특징으로 하는 정보저장매체.

청구항 21.

제20항에 있어서,

상기 삭제 정보는 상기 삭제 리스트 파일의 위치정보를 매개변수로 갖는 API(Application Program Interface)로 구현되는 것을 특징으로 하는 정보저장매체.

청구항 22.

정보저장매체에 기록된 AV 데이터를 인터랙티브 모드로 재생하는 방법에 있어서,

- (a) 독출된 마크업 문서에 포함된 프리로드 정보를 해석하는 단계;
- (b) 해석된 프리로드 정보에 따른, 심리스 재생이 보장되는 AV 데이터에 대한 프리로드 대상 파일을 호출하여 캐시 메모리에 저장하는 단계;
- (c) 상기 AV 데이터를 독출하여 버퍼 메모리에 저장하는 단계; 및

(d) 상기 버퍼 메모리로부터 상기 AV 데이터를 독출하고, 상기 캐시 메모리로부터 상기 프리로드 대상 파일을 독출하여 인터랙티브 모드로 재생하는 단계를 포함하는 것을 특징으로 하는 재생방법.

청구항 23.

제22항에 있어서,

상기 (a) 단계는

(a1) 프리로드 대상 파일의 경로 및 속성을 파악하는 단계임을 특징으로 하는 재생방법.

청구항 24.

제23항에 있어서,

상기 (a1) 단계는 헤드 태그 내에 있는 링크 태그 내에 기록된 상기 프리로드 리스트 파일의 경로를 인식하는 단계임을 특징으로 하는 재생방법.

청구항 25.

제24항에 있어서,

상기 (b) 단계는

(b1) 파악된 경로로부터 상기 프리로드 대상 파일을 독출하는 단계; 및

(b2) 독출된 프리로드 대상 파일을 파악된 속성에 따라 처리하여 저장하는 단계를 포함하는 것을 특징으로 하는 재생방법.

청구항 26.

제22항에 있어서,

상기 (a) 단계 및 (b) 단계는 바디 태그 내에 삽입되어, 프리로드 리스트 파일의 경로를 매개변수로 가지고 상기 프리로드 리스트 파일에 기록된 상기 프리로드 대상 파일을 호출하는 API를 해석하여 실행하는 것을 특징으로 하는 재생방법.

청구항 27.

제22항에 있어서,

상기 (b) 단계는 상기 정보저장매체로부터 상기 프리로드 대상 파일을 호출하는 단계를 포함하는 것을 특징으로 하는 재생방법.

청구항 28.

제22항에 있어서,

상기 (b) 단계는 인터넷 서버로부터 상기 프리로드 대상 파일을 호출하는 단계를 포함하는 것을 특징으로 하는 재생방법.

청구항 29.

정보저장매체에 기록된 AV 데이터를 인터랙티브 모드로 재생하는 장치에 있어서,

마크업 문서 또는 AV 데이터를 독출하는 독출부;

수신된 프리로드 파일 또는 AV 데이터를 저장하는 메모리;

상기 메모리에 저장된 AV 데이터를 디코딩하는 AV 디코더; 및

독출된 마크업 문서에 포함된 프리로드 정보를 해석하여 해석된 프리로드 정보에 따른, 심리스 재생이 보장되는 AV 데이터에 대한 프리로드 대상 파일이 먼저 상기 메모리에 저장되도록 요청하고 나서, 상기 AV 데이터가 독출되어 상기 메모리에 저장되도록 요청하고, 상기 메모리로부터 상기 프리로드 대상 파일을 독출하여 상기 AV 디코더로부터 출력된 AV 데이터와 함께 디스플레이하는 프리젠테이션 엔진을 포함하는 것을 특징으로 하는 재생장치.

청구항 30.

제29항에 있어서,

상기 메모리는

상기 AV 데이터를 저장하는 버퍼 메모리; 및

상기 프리로드 대상 파일을 저장하는 캐시 메모리를 포함하는 것을 특징으로 하는 재생장치.

청구항 31.

제29항에 있어서,

상기 프리젠테이션 엔진은 상기 프리로드 정보에 기초하여 프리로드 대상 파일의 경로 및 속성을 파악하고 파악된 경로로부터 프리로드 대상 파일을 호출한 다음 상기 속성에 대응되도록 상기 캐시 메모리에 저장하는 것을 특징으로 하는 재생장치.

청구항 32.

제31항에 있어서,

상기 프리젠테이션 엔진은 상기 독출부 또는 인터넷 서버로 상기 프리로드 대상 파일을 요청하는 것을 특징으로 하는 재생장치.

청구항 33.

제29항 또는 제30항에 있어서,

상기 프리로드 정보는 상기 프리로드 대상 파일의 크기정보를 더 포함하고,

상기 프리젠테이션 엔진은 상기 캐시 메모리의 잔여공간의 크기와 상기 프리로드 대상 파일의 크기를 비교하여 상기 캐시 메모리의 잔여공간의 크기가 더 작을 경우 에러이벤트를 발생시키는 것을 특징으로 하는 재생장치.

청구항 34.

제29항 또는 제30항에 있어서,

상기 위치정보는 상기 프리로드 리스트 파일의 경로로 표시되고, 상기 경로에는 상기 캐시 메모리, 상기 정보저장매체, 인터넷 서버 중 어느 하나를 가리키는 리소스 로케이터가 부가되어 있으며,

상기 프리젠테이션 엔진은 상기 프리로드 리스트 파일의 경로에 상기 캐시 메모리를 가리키는 리소스 로케이터가 부가되어 있는 경우 상기 캐시 메모리로 대응 프리로드 대상 파일을 요청하고 상기 캐시 메모리에 상기 프리로드 대상 파일이 존재하지 않을 경우 에러이벤트를 발생시키는 것을 특징으로 하는 재생장치.

청구항 35.

제29항 또는 제30항에 있어서,

상기 마크업 문서에는 삭제 대상 파일의 리스트가 기록된 삭제 리스트 파일의 위치정보가 포함되어 있고 상기 삭제 리스트에 기록된 삭제 대상 파일을 상기 메모리로부터 삭제할 것을 명령하는 삭제 정보가 더 기록되어 있고,

상기 프리젠테이션 엔진은 상기 삭제 정보를 해석하여 상기 캐시 메모리로 상기 삭제 대상 파일을 삭제할 것을 요청하는 것을 특징으로 하는 재생장치.

청구항 36.

프리로드를 수행하는 방법에 있어서,

- (a) 심리스 재생이 보장되는 AV 데이터에 대한 프리로드 대상 파일의 독출 속도를 알아내는 단계;
- (b) 대응 AV 데이터가 끊기지 않고 재생되도록 버퍼링이 수행되기 위한 조건을 알아내는 단계; 및
- (c) 알아낸 조건에 부합하는 시점에서 프리로드를 수행하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 37.

정보저장매체에 데이터를 기록하는 방법에 있어서,

- (a) 프리로드 대상 파일을 기록하는 단계;
- (b) 기록된 프리로드 대상 파일의 독출 속도를 알아내는 단계;
- (c) 대응 AV 데이터가 끊기지 않고 재생되도록 버퍼링이 수행되기 위한 조건을 알아내는 단계; 및
- (d) 알아낸 조건에 부합하는 시점에서 프리로드가 수행되기 위한 소스 코드를 기록하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 38.

프리로드를 위한 메모리 관리방법에 있어서,

- (a) 프리로드 대상 파일의 상태 정보가 포함된 메모리 관리 테이블을 기록하는 단계; 및
- (b) 상기 상태 정보를 기초로 상기 프리로드 대상 파일을 삭제하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 39.

제38항에 있어서,

(c) 상기 상태 정보를 기초로 상기 프리로드 대상 파일에 대한 가비지 콜렉션을 수행하는 단계를 더 포함하는 것을 특징으로 하는 방법.

청구항 40.

제38항 또는 제39항에 있어서,

상기 (b) 단계는 해당 프리로드 대상 파일에 대한 사용이 완료되면 사용 중이 아님과 삭제가능함을 상기 상태 정보로서 표시하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 41.

제38항 또는 제39항에 있어서,

상기 (c) 단계는

(c1) 해당 프리로드 대상 파일이 사용 중이 아니고 삭제가능한 경우 해당 프리로드 파일을 물리적으로 삭제하는 단계 ; 및

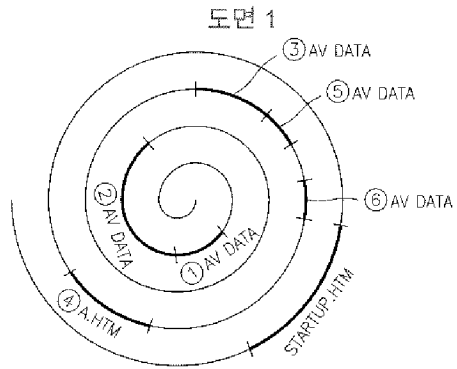
(c2) 삭제된 프리로드 대상 파일이 삭제되어 존재하지 않음을 상기 상태 정보로서 표시하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 42.

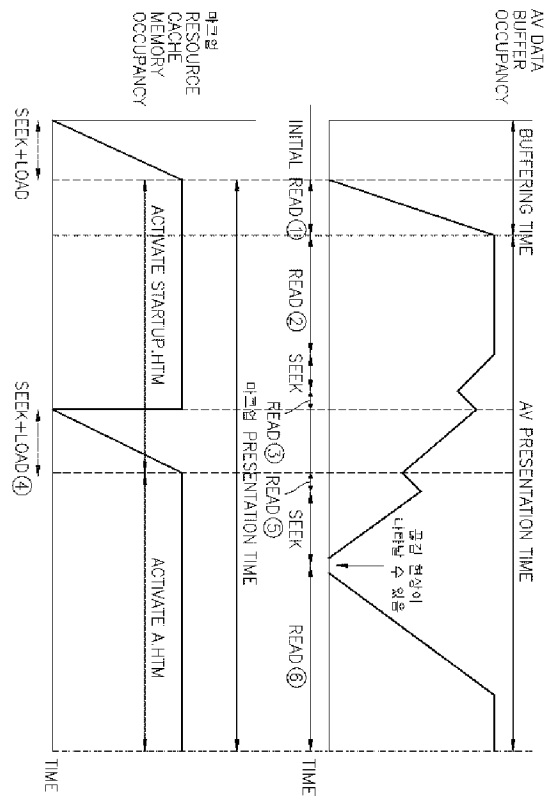
제41항에 있어서,

상기 (c) 단계는

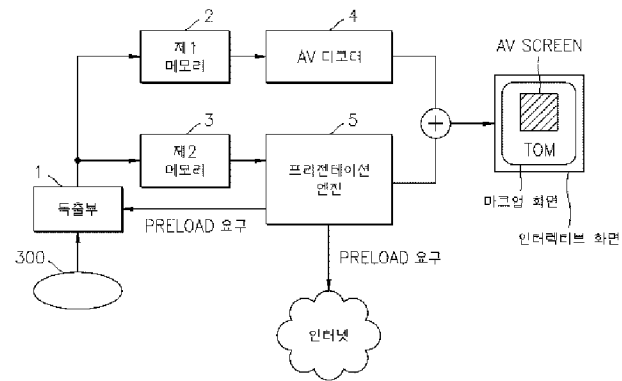
(c3) 메모리에 기록된 다른 프리로드 대상 파일을 재배열하는 단계를 포함하는 것을 특징으로 하는 방법.



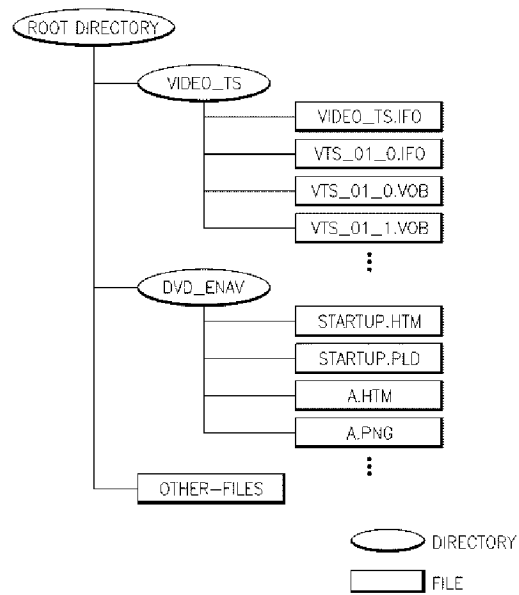
도면 2



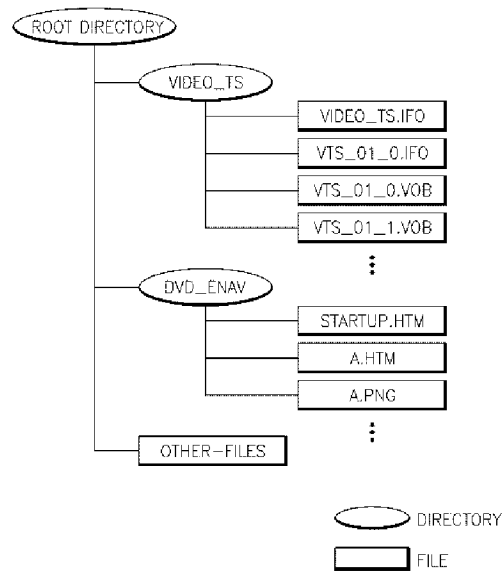
도면 3



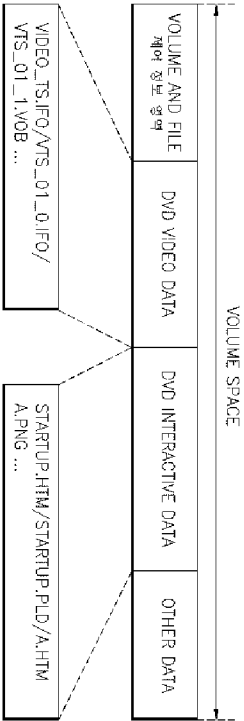
도면 4a



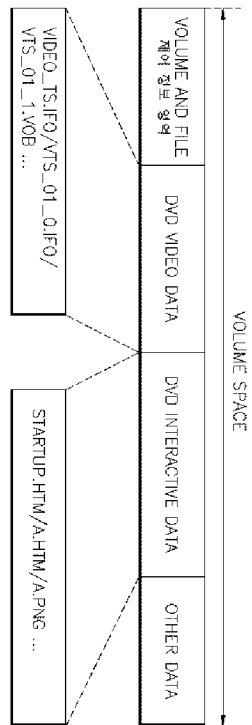
도면 4b

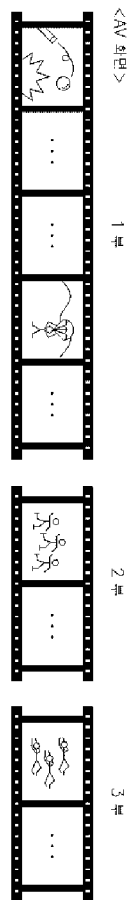


도면 5a

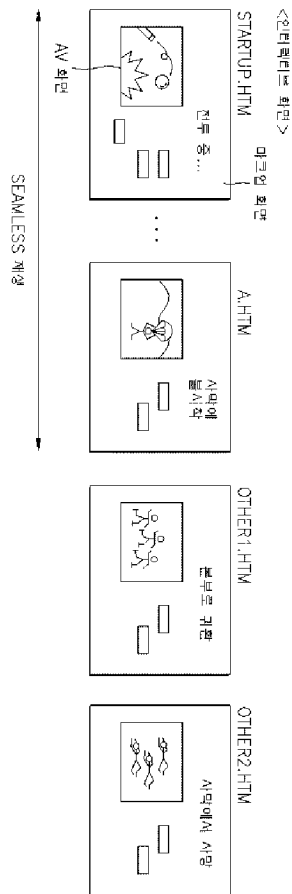


도면 5b

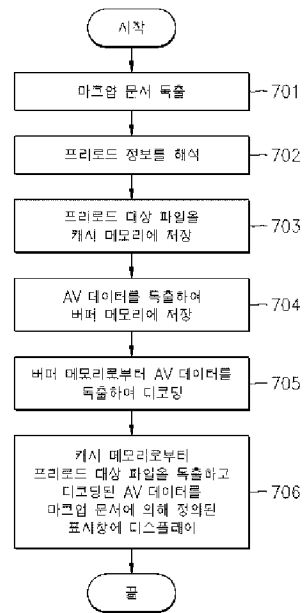




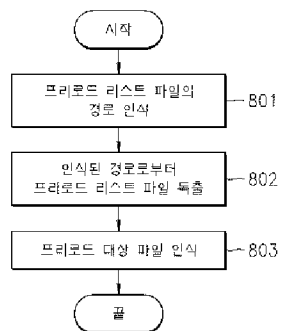
도면



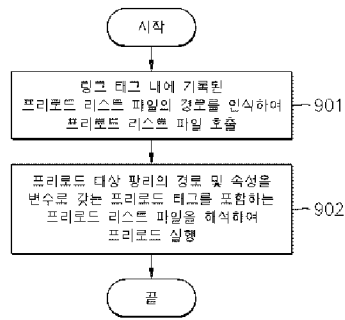
도면 7



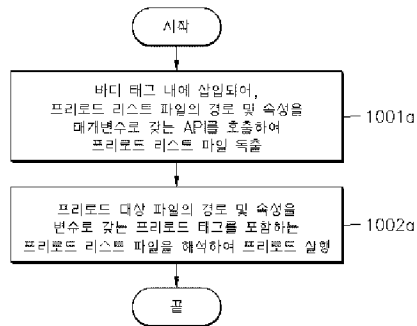
도면 8



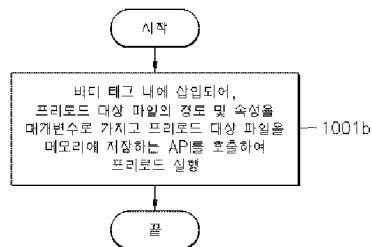
도면 9



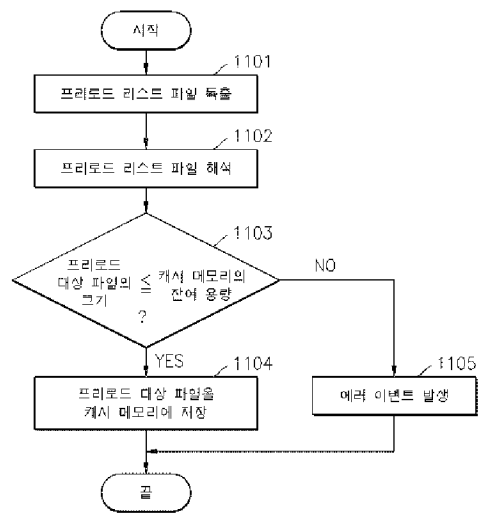
도면 10a



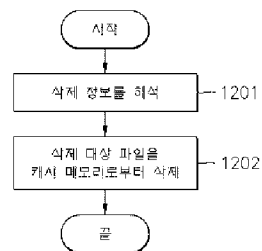
도면 10b



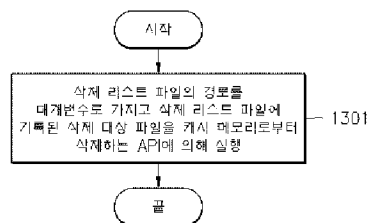
도면 11



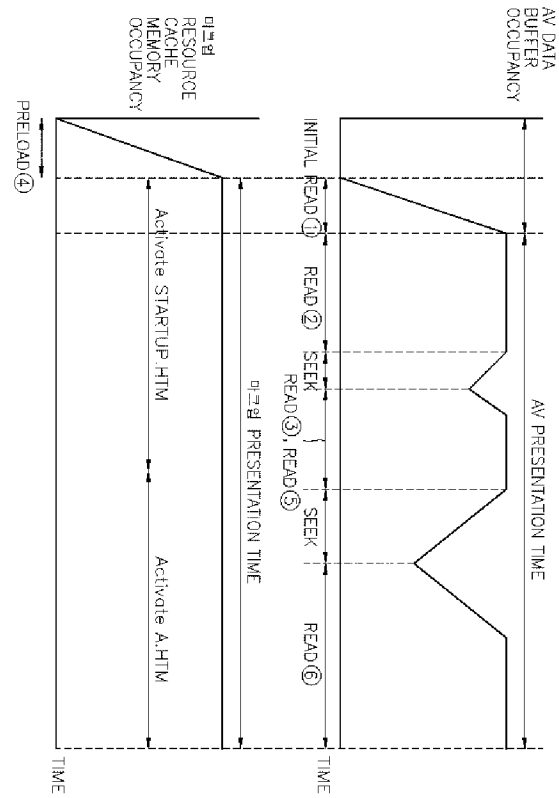
도면 12



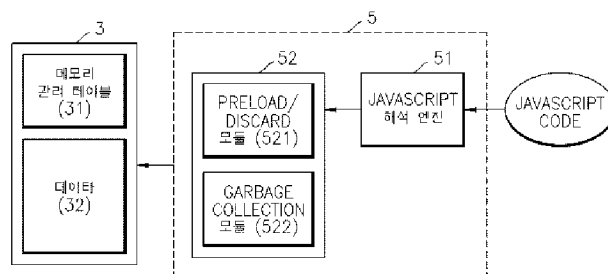
도면 13



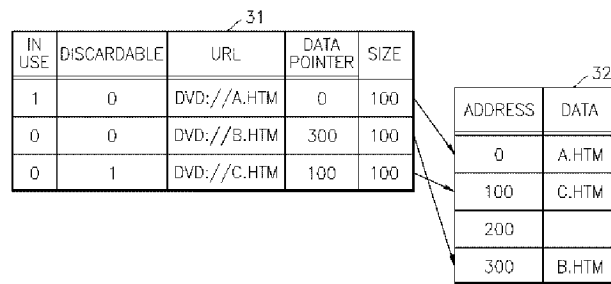
도면 14



도면 15

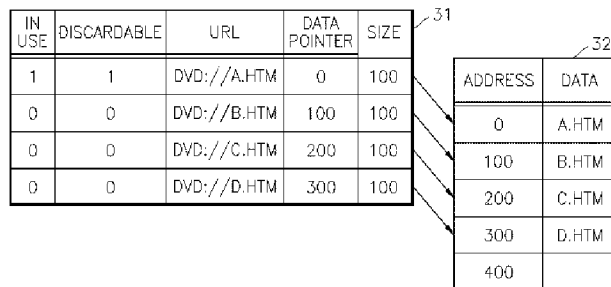


도면 16



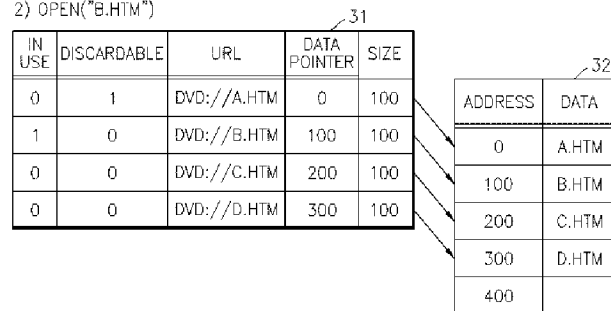
도면 17a

1) OPEN("A.HTM"), PRELOAD("B.HTM"), PRELOAD("C.HTM"), PRELOAD("D.HTM")



도면 17b

2) OPEN("B.HTM")



도면 17c

3) GARBAGE COLLECTION

IN USE	DISCARDABLE	URL	31		ADDRESS	DATA
			DATA POINTER	SIZE		
0	1	DVD://A.HTM	-1	100	0	B.HTM
1	0	DVD://B.HTM	0	100	100	C.HTM
0	0	DVD://C.HTM	100	100	200	D.HTM
0	0	DVD://D.HTM	200	100	300	
					400	

도면 17d

4) OPEN("F.HTM")

IN USE	DISCARDABLE	URL	31		ADDRESS	DATA
			DATA POINTER	SIZE		
1	1	DVD://F.HTM	300	100	0	B.HTM
0	0	DVD://B.HTM	0	100	100	C.HTM
0	0	DVD://C.HTM	100	100	200	D.HTM
0	0	DVD://D.HTM	200	100	300	F.HTM
					400	

도면 17e

5) DISCARD("A")

IN USE	DISCARDABLE	URL	31		ADDRESS	DATA
			DATA POINTER	SIZE		
1	1	DVD://F.HTM	300	100	0	B.HTM
0	1	DVD://B.HTM	0	100	100	C.HTM
0	1	DVD://C.HTM	100	100	200	D.HTM
0	1	DVD://D.HTM	200	100	300	F.HTM
					400	

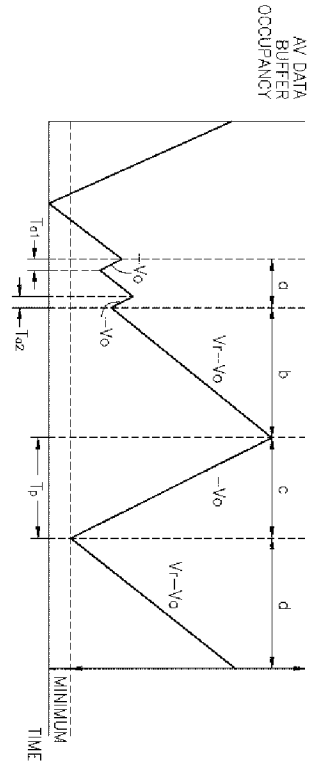
도면 17f

6) GARBAGE COLLECTION

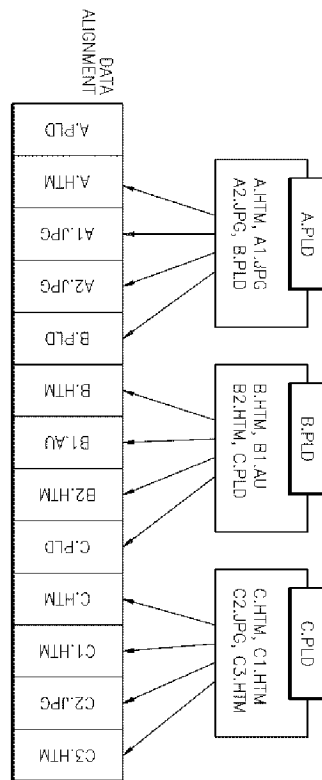
IN USE	DISCARDABLE	URL	DATA POINTER	SIZE
1	1	DVD://F.HTM	0	100
0	1	DVD://B.HTM	-1	100
0	1	DVD://C.HTM	-1	100
0	1	DVD://D.HTM	-1	100

ADDRESS	DATA
0	F.HTM
100	
200	
300	
400	

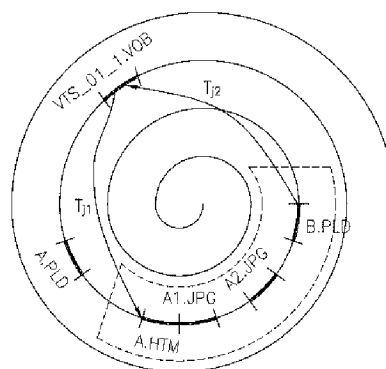
도면 18



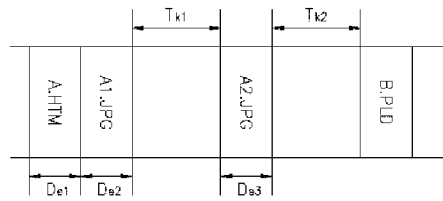
도면 19



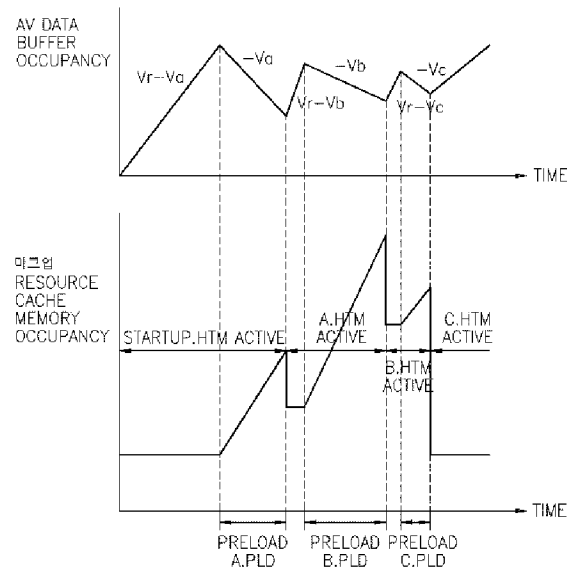
도면 20a



도면 20b



도면 21



도면 22

